Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1951	707/2.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:31
L2	167	optimiz$5 and "logical partition" and processor$1 and dynamic$5 and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:33
L3	14158	search$3 and quer$3 same database$1 and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:34
L4	40	2 and 3	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:34
L5	306	(software or program$1) and "logical partition" and processor$1 and dynamic$5 and @ad<"20010101"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:36
L6	42	(first same (software or program$1)) and (second same (software or program$1) same processor$1) and "logical partition" and processor$1 and dynamic$5 and @ad<"20010101"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:37
L7	42	5 and 6	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:37
L8	7948	(search$3 or query$3 or queries) same (optimis$6 or optimiz$6) and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:39
L9	805	"707"/$.ccls. and ((query$4 or queries) near execut$4) and (partition$4 or cluster$4)	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:42
L10	329	8 and 9	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:42
L11	127	(first same (software or program$1)) and (second same (software or program$1)) and "logical partition" and processor$1 and dynamic$5 and @ad<"20010101"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:43

L12	3	10 and 11	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:43
L13	10	first same search$3 same strategy same processor$1 and resource$1 and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:45
L14	3	first same search$3 same strategy same processor$1 and resource$1 and assignment$1 and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:45
L15	76	invok$3 near1 quer$3 and search$3 and partition$1 and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:46
L16	38	"logical partition" and processor$1 and dynamic$5 same (alter$2 or chang$2) and quer$3 and search$3 and strategy and @ad<"20011025"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:47
L17	0	15 and 16	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/03/22 10:47

Terms used **optimize** and **logical** **partion** and **processor** Found **69,108** of **151,219**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Value-based clock gating and operation packing: dynamic strategies for improving processor power and performance

David Brooks, Margaret Martonosi

May 2000 **ACM Transactions on Computer Systems (TOCS)**, Volume 18 Issue 2

Full text available: pdf(210.51 KB) Additional Information: full citation, abstract, references, citings, index terms

The large address space needs of many current applications have pushed processor designs toward 64-bit word widths. Although full 64-bit addresses and operations are indeed sometimes needed, arithmetic operations on much smaller quantities are still more common. In fact, another instruction set trend has been the introduction of instructions geared toward subword operations on 16-bit quantities. For examples, most major processors now include instruction set support for multimedia operation ...

2 The program decision logic approach to predicated execution

David I. August, John W. Sias, Jean-Michel Puiatti, Scott A. Mahlke, Daniel A. Connors, Kevin M. Crozier, Wen-mei W. Hwu

May 1999 **ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture**, Volume 27 Issue 2

Full text available: pdf(215.39 KB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Modern compilers must expose sufficient amounts of Instruction-Level Parallelism (ILP) to achieve the promised performance increases of superscalar and VLIW processors. One of the major impediments to achieving this goal has been inefficient programmatic control flow. Historically, the compiler has translated the programmer's original control structure directly into assembly code with conditional branch instructions. Eliminating inefficiencies in handling branch instructions and exploiting ILP h ...

3 A method to derive application-specific embedded processing cores

Olivier Hébert, Ivan C. Kraljic, Yvon Savaria

May 2000 **Proceedings of the eighth international workshop on Hardware/software codesign**

Full text available: pdf(101.67 KB) Additional Information: full citation, abstract, references, citings, index terms

The concept of system-on-a-chip is becoming increasingly popular for the integration of complex systems. New types of processor cores are now available that enable the designer

to customize their processors for the target applications. These soft cores are not tightly coupled with the target application, and this leads to processing cores sub-optimal for their specific applications. This paper proposes a method to derive application-specific embedded processors from soft processor cores. Th ...

Keywords: configurable processor, custom core, embedded core, soft core, system-on-a-chip

4 New topics in logic synthesis: On-chip logic minimization

Roman Lysecky, Frank Vahid

June 2003 **Proceedings of the 40th conference on Design automation**

Full text available: pdf(254.04 KB) Additional Information: full citation, abstract, references, citings, index terms

While Boolean logic minimization is typically used in logic synthesis, logic minimization can be useful in numerous other applications. However, many of those applications, such as Internet Protocol routing table and network access control list reduction, require logic minimization during the application's runtime, and hence could benefit from minimization executing on-chip alongside the application. On-chip minimization can even enable dynamic hardware/software partitioning. We discuss requirem ...

Keywords: dynamic optimization, embedded systems, logic minimization, on-chip logic minimization, on-chip synthesis, system-on-a-chip

5 A parallel logic programming approach to combinatorial optimization in design

Jim Butler, Hideomi Ohtsubo

June 1990 **Proceedings of the third international conference on Industrial and engineering applications of artificial intelligence and expert systems - Volume 2**

Full text available: pdf(571.46 KB) Additional Information: full citation, abstract, references, index terms

An approach to the solution of combinatorial optimization problems using parallel logic programming is presented. The solution space is divided with a separate process (worker process) assigned to search each subspace. In order to have information regarding the most recent optimal combination available to all worker processes, a tree of manager processes is established to relay discoveries at a given worker process to other worker processes. The performance of this method is tested ...

6 Using precomputation in architecture and logic resynthesis

Soha Hassoun, Carl Ebeling

November 1998 **Proceedings of the 1998 IEEE/ACM international conference on Computer-aided design**

Full text available: pdf(917.99 KB) Additional Information: full citation, references, citings, index terms

7 Automatically proving the correctness of compiler optimizations

Sorin Lerner, Todd Millstein, Craig Chambers

May 2003 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation**, Volume 38 Issue 5

Full text available: pdf(265.63 KB) Additional Information: full citation, abstract, references, citings, index terms

We describe a technique for automatically proving compiler optimizations *sound*, meaning that their transformations are always semantics-preserving. We first present a domain-

specific language, called Cobalt, for implementing optimizations as guarded rewrite rules. Cobalt optimizations operate over a C-like intermediate representation including unstructured control flow, pointers to local variables and dynamically allocated memory, and recursive procedures. Then we describe a technique for ...

Keywords: automated correctness proofs, compiler optimization

8 Abstract specialization and its applications

Germán Puebla, Manuel Hermenegildo

June 2003 **ACM SIGPLAN Notices , Proceedings of the 2003 ACM SIGPLAN workshop on Partial evaluation and semantics-based program manipulation**, Volume 38 Issue 10

Full text available: pdf(376.42 KB) Additional Information: full citation, abstract, references, index terms

The aim of program specialization is to optimize programs by exploiting certain knowledge about the context in which the program will execute. There exist many program manipulation techniques which allow specializing the program in different ways. Among them, one of the best known techniques is *partial evaluation*, often referred to simply as program specialization, which optimizes programs by specializing them for (partially) known input data. In this work we describe *abstract speciali ...*

***Keywords**: abstract interpretation, logic programming, partial evaluation, program optimization, program parallelization, program specialization, static analysis*

9 Energy-driven integrated hardware-software optimizations using SimplePower

N. Vijaykrishnan, M. Kandemir, M. J. Irwin, H. S. Kim, W. Ye

May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2

Full text available: pdf(314.76 KB) Additional Information: full citation, abstract, references, citings, index terms

With the emergence of a plethora of embedded and portable applications, energy dissipation has joined throughput, area, and accuracy/precision as a major design constraint. Thus, designers must be concerned with both optimizing and estimating the energy consumption of circuits, architectures, and software. Most of the research in energy optimization and/or estimation has focused on single components of the system and has not looked across the interacting spectrum of the hardware and softwar ...

Keywords: compiler optimizations, energy optimization and estimation, energy simulator, hardware-software interaction, low-power architectures, system energy

10 Energy-conscious HW/SW-partitioning of embedded systems: a case study on an MPEG-2 encoder

Jouml;rg Henkel, Yanbing Li

March 1998 **Proceedings of the 6th international workshop on Hardware/software codesign**

Full text available: pdf(234.20 KB) Publisher Site Additional Information: full citation, references, citings, index terms

11

Multiprocessor SoC MPSoC solutions/nightmare: Flexible architectures for engineering successful SOCs

Chris Rowen, Steve Leibson
June 2004 **Proceedings of the 41st annual conference on Design automation**

Full text available: pdf(96.22 KB) Additional Information: full citation, abstract, index terms

This paper focuses on a particular SOC design technology and methodology, here called the advanced or processor-centric SOC design method, which reduces the risk of SOC design and increases ROI by using configurable processors to implement on-chip functions while increasing the SOC's flexibility through software programmability. The essential enabler for this design methodology is automatic processor generation-the rapid and easy creation of new microprocessor architectures, complete with effici ...

Keywords: MPSOC, RISC, RTL, SOC, processor cores

12 Automated soundness proofs for dataflow analyses and transformations via local rules

Sorin Lerner, Todd Millstein, Erika Rice, Craig Chambers
January 2005 **Proceedings of the 32nd ACM SIGPLAN-SIGACT sysposium on Principles of programming languages**

Full text available: pdf(262.33 KB) Additional Information: full citation, abstract, references, index terms

We present Rhodium, a new language for writing compiler optimizations that can be automatically proved sound. Unlike our previous work on Cobalt, Rhodium expresses optimizations using explicit dataflow facts manipulated by local propagation and transformation rules. This new style allows Rhodium optimizations to be mutually recursively defined, to be automatically composed, to be interpreted in both flow-sensitive and -insensitive ways, and to be applied interprocedurally given a separate contex ...

Keywords: automated correctness proofs, compiler optimization

13 Data base processor mage

G. Berger Sabbatel
March 1980 **Proceedings of the fifth workshop on Computer architecture for non-numeric processing**

Full text available: pdf(641.92 KB) Additional Information: full citation, abstract, references, index terms

In this paper, we present the design of data base processor MAGE. This DBP is based on a hierarchical DB access method. It is composed of two microprocessors, a disk processor and a moving head disk. The processor requirements, design decisions, and architecture are discussed. We start with an overview of DBP framework. The MAGE DB access method is then summarized. Finally, we present the design and implementation of the DBP, with a particular emphasis on the disk processor.

14 Reconfigurable computing: a survey of systems and software

Katherine Compton, Scott Hauck
June 2002 **ACM Computing Surveys (CSUR)**, Volume 34 Issue 2

Full text available: pdf(710.56 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Due to its potential to greatly accelerate a wide variety of applications, reconfigurable computing has become a subject of a great deal of research. Its key feature is the ability to perform computations in hardware to increase performance, while retaining much of the flexibility of a software solution. In this survey, we explore the hardware aspects of reconfigurable computing machines, from single chip architectures to multi-chip systems, including internal structures and external coupling. W ...

Keywords: Automatic design, FPGA, field-programmable, manual design, reconfigurable architectures, reconfigurable computing, reconfigurable systems

15 Architectures: Opportunities and challenges in application-tuned circuits and architectures based on nanodevices

Teng Wang, Zhenghua Qi, Csaba Andras Moritz

April 2004 **Proceedings of the first conference on computing frontiers on Computing frontiers**

Full text available: pdf(288.94 KB) Additional Information: full citation, abstract, references, index terms

Nanoelectronics research has primarily focused on devices. By contrast, not much has been published on innovations at higher layers: we know little about how to construct circuits or architectures out of such devices. In this paper, we focus on the currently most promising nanodevice technologies, such as arrays of semiconductor nanowires (NWs) and arrays of crossed carbon nanotubes (CNTs). In contrast to general-purpose programmable fabrics (such as PLAs), we investigate nano-fabrics that, whil ...

Keywords: FET, NASIC, microwire, nanowire, tile

16 Design innovations for embedded processors: A fast on-chip profiler memory

Roman Lysecky, Susan Cotterell, Frank Vahid

June 2002 **Proceedings of the 39th conference on Design automation**

Full text available: pdf(277.15 KB) Additional Information: full citation, abstract, references, citings, index terms

Profiling an application executing on a microprocessor is part of the solution to numerous software and hardware optimization and design automation problems. Most current profiling techniques suffer from runtime overhead, inaccuracy, or slowness, and the traditional non-intrusive method of using a logic analyzer doesn't work for today's system-on-a-chip having embedded cores. We introduce a novel on-chip memory architecture that overcomes these limitations. The architecture, which we call ProMem ...

Keywords: adaptive architectures, binary tree, embedded CAD, embedded systems, low power, memory design, platform tuning, profiling, system-on-a-chip

17 A schema for interprocedural modification side-effect analysis with pointer aliasing

Barbara G. Ryder, William A. Landi, Philip A. Stocks, Sean Zhang, Rita Altucher

March 2001 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 23 Issue 2

Full text available: pdf(1.72 MB) Additional Information: full citation, abstract, references, citings, index terms, review

The first interprocedural modification side-effects analysis for C (MODC) that obtains better than worst-case precision on programs with general-purpose pointer usage is presented with empirical results. The analysis consists of an algorithm schema corresponding to a family of MODC algorithms with two independent phases: one for determining pointer-induced aliases and a subsequent one for propagating interprocedural ...

18 Simple relational correctness proofs for static analyses and program transformations

Nick Benton

January 2004 **ACM SIGPLAN Notices , Proceedings of the 31st ACM SIGPLAN-SIGACT symposium on Principles of programming languages**, Volume 39 Issue 1

Full text available: pdf(139.63 KB) Additional Information: full citation, abstract, references, citings, index terms

We show how some classical static analyses for imperative programs, and the optimizing

transformations which they enable, may be expressed and proved correct using elementary logical and denotationaltechniques. The key ingredients are an interpretation of program properties as relations, rather than predicates, and a realization that although many program analyses are traditionally formulated in very intensional terms, the associated transformations are actually enabled by more liberal extension ...

Keywords: Hoare logic, denotational semantics, dependency, information flow, optimizing compilation, partial equivalence relations, program analysis, security, types

19 Energy-effective issue logic

Daniele Folegnani, Antonio González
May 2001 **ACM SIGARCH Computer Architecture News , Proceedings of the 28th annual international symposium on Computer architecture**, Volume 29 Issue 2

Full text available: pdf(774.80 KB) Additional Information: full citation, abstract, references, citings, index terms

The issue logic of a dynamically-scheduled superscalar processor is a complex mechanism devoted to start the execution of multiple instructions every cycle. Due to its complexity, it is responsible for a significant percentage of the energy consumed by a microprocessor. The energy consumption of the issue logic depends on several architectural parameters, the instruction issue queue size being one of the most important. In this paper we present a technique to reduce the energy consumption ...

***Keywords**: adaptive hardware, energy consumption, issue logic, low power*

20 Warp, an integrated solution of high-speed parallel computing

S. Borkar, R. Cohn, G. Cox, S. Gleason, T. Gross
November 1988 **Proceedings of the 1988 ACM/IEEE conference on Supercomputing**

Full text available: pdf(1.35 MB) Additional Information: full citation, abstract, references, citings, index terms

iWarp is a system architecture for high speed signal, image and scientific computing. The heart of an iWarp system is the iWarp component: a single chip processor that requires only the addition of memory chips to form a complete system building block, called the iWarp cell. Each iWarp component contains both a powerful computation engine (20 MFLOPS) and a high throughput (320 MBytes/sec), low latency (100-150 ns) communication engine for interfacing with other iWarp cells. Because of its s ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used
search strategy and **resouce assignment** and **save** and **persistent object**

Found **30,836** of **151,219**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Special issue on persistent object systems: Adaptable pointer swizzling strategies in object bases: design, realization, and quantitative analysis
Alfons Kemper, Donald Kossmann
July 1995 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 4 Issue 3
Full text available: pdf(2.69 MB) Additional Information: full citation, abstract, references, citings

In this article, different techniques for *"pointer swizzling"* are classified and evaluated for optimizing the access to main-memory resident persistent objects. To speed up the access along inter-object references, the persistent pointers in the form of unique object identifiers (OIDs) are transformed (swizzled) into main-memory pointers (addresses). Pointer swizzling techniques can be divided into two classes: (1) those that allow replacement of swizzled objects from the buffer before th ...

Keywords: object-oriented database systems, performance evaluation, pointer swizzling

2 Special issue on persistent object systems: Tigukat: a uniform behavioral objectbase management system
M. Tamer Özsu, Randal Peters, Duane Szafron, Boman Irani, Anna Lipka, Adriana Muñoz
July 1995 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 4 Issue 3
Full text available: pdf(2.78 MB) Additional Information: full citation, abstract, references, citings

We describe the TIGUKAT objectbase management system, which is under development at the Laboratory for Database Systems Research at the University of Alberta. TIGUKAT has a novel object model, whose identifying characteristics include a purely behavioral semantics and a uniform approach to objects. Everything in the system, including types, classes, collections, behaviors, and functions, as well as meta-information, is a first-class object with well-defined behavior. In this way, the model abstr ...

Keywords: database management, objectbase management, persistent storage system, reflective system

3 Query evaluation techniques for large databases
Goetz Graefe
June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2

Full text available: pdf(9.37 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

4 Fast detection of communication patterns in distributed executions

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

5 Special issue on prototypes of deductive database systems: The CORAL deductive system

Raghu Ramakrishnan, Divesh Srivastava, S. Sudarshan, Praveen Seshadri

April 1994 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 3 Issue 2

Full text available: pdf(3.03 MB) Additional Information: full citation, abstract, references, citings

CORAL is a deductive system that supports a rich declarative language, and an interface to C++, which allows for a combination of declarative and imperative programming. A CORAL declarative program can be organized as a collection of interacting modules. CORAL supports a wide range of evaluation strategies, and automatically chooses an efficient strategy for each module in the program. Users can guide query optimization by selecting from a wide range of control choices. The CORAL system provides ...

Keywords: deductive database, logic programming system, query language

6 Types and persistence in database programming languages

Malcolm P. Atkinson, O. Peter Buneman

June 1987 **ACM Computing Surveys (CSUR)**, Volume 19 Issue 2

Full text available: pdf(7.91 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Traditionally, the interface between a programming language and a database has either been through a set of relatively low-level subroutine calls, or it has required some form of embedding of one language in another. Recently, the necessity of integrating database and programming language techniques has received some long-overdue recognition. In

response, a number of attempts have been made to construct programming languages with completely integrated database management systems. These lang ...

7 Object-oriented technology: TIGUKAT object management system: initial design and current directions

M. Tamer Özsu, Randal Peters, Boman Irani, Anna Lipka, Adriana Munoz, Duane Szafron

October 1993 **Proceedings of the 1993 conference of the Centre for Advanced Studies on Collaborative research: software engineering - Volume 1**

Full text available: pdf(1.53 MB) Additional Information: full citation, abstract, references

We describe the TIGUKAT object management system that is under development at the Laboratory for Database Systems Research of the University of Alberta. TIGUKAT has a novel object model whose identifying characteristics include a purely behavioral semantics and a uniform approach to objects. Everything in the system is a first-class object with well-defined behavior. The computational model supported is one of applying behaviors to objects. A query model has been developed for TIGUKAT that is co ...

8 Special system-oriented section: the best of SIGMOD '94: QuickStore: a high performance mapped object store

Seth J. White, David J. DeWitt

October 1995 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 4 Issue 4

Full text available: pdf(2.58 MB) Additional Information: full citation, abstract, references, citings

QuickStore is a memory-mapped storage system for persistent C++, built on top of the EXODUS Storage Manager. QuickStore provides fast access to in-memory objects by allowing application programs to access objects via normal virtual memory pointers. This article presents the results of a detailed performance study using the OO7 benchmark. The study compares the performance of QuickStore with the latest implementation of the E programming language. The QuickStore and E systems exemplify the two ba ...

Keywords: benchmark, client-server, memory-mapped, object-oriented, performance, pointer swizzling

9 The treatment of state in optimistic systems

David Bruce

July 1995 **ACM SIGSIM Simulation Digest , Proceedings of the ninth workshop on Parallel and distributed simulation**, Volume 25 Issue 1

Full text available: pdf(1.59 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Optimistic computation methods typically save copies of objects' state information, so that they can recover from erroneous "over-optimistic" computations. Such state saving is generally time and space consuming, and can be rather complicated both to implement and to use.I show how the data structure community's theory of persistence can be used not only to analyse and explain the treatment of state in optimistic systems, but also as a simple yet general mechanism for ...

Keywords: optimistic methods, parallel discrete event simulation, persistent data structures, state saving

10 A declarative approach to optimize bulk loading into databases

Sihem Amer-Yahia, Sophie Cluet

June 2004 **ACM Transactions on Database Systems (TODS)**, Volume 29 Issue 2

Full text available: pdf(1.00 MB) Additional Information: full citation, abstract, references, index terms

Applications, such as warehouse maintenance, need to load large data volumes regularly. The efficiency of loading depends on the resources that are available at the source and at the target systems. Our work aims to understand the performance criteria that are involved in bulk loading data into a database and to devise tailored optimization strategies.Unlike commercial systems and previous research on the same topic, our approach follows the fundamental database principle of physical-logical ind ...

Keywords: Declarative bulk loading, algebra, recovery, side-effects

11 Garbage collection for a client-server persistent object store

Laurent Amsaleg, Michael J. Franklin, Olivier Gruber

August 1999 **ACM Transactions on Computer Systems (TOCS)**, Volume 17 Issue 3

Full text available: pdf(267.18 KB) Additional Information: full citation, abstract, references, citings, index terms, review

We describe an efficient server-based algorithm for garbage collecting persistent object stores in a client-server environmnet. The algorithm is incremental and runs concurrently with client transactions. Unlike previous algorithms, it does not hold any transactional locks on data and does non require callbacks to clients. It is fault-tolerant, but performs very little logging. The algorithm has been designed to be integrated into existing systems, and therefore it works with standard i ...

Keywords: client-server system, logging, persistent object-store, recovery

12 On semantic caching and query scheduling for mobile nearest-neighbor search

Baihua Zheng, Wang-Chien Lee, Dik Lun Lee

November 2004 **Wireless Networks**, Volume 10 Issue 6

Full text available: pdf(293.57 KB) Additional Information: full citation, abstract, references, index terms

Location-based services have received increasing attention in recent years. In this paper, we address the performance issues of mobile nearest-neighbor search, in which the mobile user issues a query to retrieve stationary service objects nearest to him/her. An index based on Voronoi Diagram is used in the server to support such a search, while a semantic cache is proposed to enhance the access efficiency of the service. Cache replacement policies tailored for the proposed semantic cache are ...

Keywords: Voronoi diagram, indexing technique, location-based services, nearest-neighbor search, query scheduling, roaming, semantic caching

13 Process migration

September 2000 **ACM Computing Surveys (CSUR)**, Volume 32 Issue 3

Full text available: pdf(1.24 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Process migration is the act of transferring a process between two machines. It enables dynamic load distribution, fault resilience, eased system administration, and data access locality. Despite these goals and ongoing research efforts, migration has not achieved widespread use. With the increasing deployment of distributed systems in general, and distributed operating systems in particular, process migration is again receiving more attention in both research and product development. As hi ...

Keywords: distributed operating systems, distributed systems, load distribution, process

migration

14 Affinity-based management of main memory database clusters
Minwen Ji
November 2002 **ACM Transactions on Internet Technology (TOIT)**, Volume 2 Issue 4
Full text available: pdf(553.96 KB) Additional Information: full citation, abstract, references, index terms

We study management strategies for main memory database clusters that are interposed between Internet applications and back-end databases as content caches. The task of management is to allocate data across individual cache databases and to route queries to the appropriate databases for execution. The goal is to maximize effective cache capacity and to minimize synchronization cost. We propose an affinity-based management system for main memory database cLUsters (*ALBUM*). ALBUM executes ea ...

Keywords: Main memory database, clustering, database administration, database cluster, file organization, query affinity, scalability

15 Focus and Context: Popout prism: adding perceptual principles to overview+detail document interfaces
Bongwon Suh, Allison Woodruff, Ruth Rosenholtz, Alyssa Glass
April 2002 **Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves**
Full text available: pdf(1.56 MB) Additional Information: full citation, abstract, references, citings, index terms

We present an overview+detail document interface that draws on perceptual principles to help users work with documents. Central to our approach is the use of improved document overviews. Our approach also includes novel highlighting in the full representation of documents, as well as techniques to help users smoothly transition from the overview to the full representation of the document. We present a specific implementation of our design for Web browsing. We also present a qualitative user stud ...

Keywords: document interfaces, overview+detail, popout effects, thumbnails

16 A novel application development environment for large-scale scientific computations
X. Shen, W. Liao, A. Choudhary, G. Memik, M. Kandemir, S. More, G. Thiruvathukal, A. Singh
May 2000 **Proceedings of the 14th international conference on Supercomputing**
Full text available: pdf(1.15 MB) Additional Information: full citation, abstract, references, citings, index terms

Our results demonstrate that our novel application development environment provides both ease-of-use and high performance for large-scale, I/O-intensive scientific applications.

17 An optimized implementation for VML based on pattern matching and dynamic programming
Weimin Chen, Volker Turau
November 1994 **Proceedings of the third international conference on Information and knowledge management**
Full text available: pdf(930.47 KB) Additional Information: full citation, abstract, references, index terms

In an object-oriented database system (OODBS), objects exist persistently and object I/O is transparent to the programmer. Therefore, some mechanism in the system must initiate I/O as the program runs. In this paper we present an approach based on pattern matching and

dynamic programming that allows a program to interact efficiently with the runtime storage layer. We are interested in allowing programs to manipulate very large objects without necessarily reading them entirely. If a program ...

18 Reactive tabu search in unmanned aerial reconnaissance simulations
Joel L. Ryan, T. Glenn Bailey, James T. Moore, William B. Carlton
December 1998 **Proceedings of the 30th conference on Winter simulation**
Full text available: pdf(117.00 KB) Additional Information: full citation, references, index terms

19 Experience with a sofiware-defined machine architecture
David W. Wall
May 1992 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 14 Issue 3
Full text available: pdf(2.86 MB) Additional Information: full citation, abstract, references, citings, index terms, review

We have built a system in which the compiler back end and the linker work together to present an abstract machine at a considerably higher level than the actual machine. The intermediate language translated by the back end is the target language of all high-level compilers and is also the only assembly language generally available. This lets us do intermodule register allocation, which would be harder if some of the code in the program had come from a traditional assembler, out of sight of ...

Keywords: RISC, graph coloring, intermediate language, interprocedural, optimization, pipeline scheduling, profiling, register allocation, register windows

20 The design of POSTGRES
Michael Stonebraker, Lawrence A. Rowe
June 1986 **ACM SIGMOD Record , Proceedings of the 1986 ACM SIGMOD international conference on Management of data**, Volume 15 Issue 2
Full text available: pdf(1.91 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents the preliminary design of a new database management system, called POSTGRES, that is the successor to the INGRES relational database system. The main design goals of the new system are to provide better support for complex objects, provide user extendibility for data types, operators and access methods, provide facilities for active databases (i.e., alerters and triggers) and inferencing including forward- ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used
invoke and **search** **strategy** and **execution** and **logical** **partition**

Found **52,798** of **151,219**

Sort results by: relevance
Display results: expanded form

Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown

Relevance scale

1 Fast detection of communication patterns in distributed executions
Thomas Kunz, Michiel F. H. Seuren
November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**
Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

2 Query evaluation techniques for large databases
Goetz Graefe
June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2
Full text available: pdf(9.37 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

3 Parallel execution of prolog programs: a survey
Gopal Gupta, Enrico Pontelli, Khayri A.M. Ali, Mats Carlsson, Manuel V. Hermenegildo
July 2001 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 23 Issue 4
Full text available: pdf(1.95 MB) Additional Information: full citation, abstract, references, citings, index

terms

Since the early days of logic programming, researchers in the field realized the potential for exploitation of parallelism present in the execution of logic programs. Their high-level nature, the presence of nondeterminism, and their referential transparency, among other characteristics, make logic programs interesting candidates for obtaining speedups through parallel execution. At the same time, the fact that the typical applications of logic programming frequently involve irregular computatio ...

Keywords: Automatic parallelization, constraint programming, logic programming, parallelism, prolog

4 PARLOG: parallel programming in logic 

Keith Clark, Steve Gregory

January 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 8 Issue 1

Full text available: pdf(3.79 MB) Additional Information: full citation, abstract, references, citings, index terms, review

PARLOG is a logic programming language in the sense that nearly every definition and query can be read as a sentence of predicate logic. It differs from PROLOG in incorporating parallel modes of evaluation. For reasons of efficient implementation, it distinguishes and separates and-parallel and or-parallel evaluation. PARLOG relations are divided into two types: single-solution relations and all-solutions relations. A conjunction of single-solution relation calls can be evaluated ...

5 A software engineering perspective on algorithmics 

Karsten Weihe

March 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 1

Full text available: pdf(1.62 MB) Additional Information: full citation, abstract, references, index terms, review

An algorithm component is an implementation of an algorithm which is not intended to be a stand-alone module, but to perform a specific task within a large software package or even within several distinct software packages. Therefore, the design of algorithm components must also incorporate software-engineering aspects. A key design goal is adaptability. This goal is important for maintenance throughout a project, prototypical development, and reuse in new, unforseen contex ...

Keywords: algorithm engineering

6 Parallel logic simulation of VLSI systems 

Mary L. Bailey, Jack V. Briner, Roger D. Chamberlain

September 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 3

Full text available: pdf(3.74 MB) Additional Information: full citation, abstract, references, citings, index terms

Fast, efficient logic simulators are an essential tool in modern VLSI system design. Logic simulation is used extensively for design verification prior to fabrication, and as VLSI systems grow in size, the execution time required by simulation is becoming more and more significant. Faster logic simulators will have an appreciable economic impact, speeding time to market while ensuring more thorough system design testing. One approach to this problem is to utilize parallel processing, taking ...

Keywords: circuit structure, parallel architecture, parallelism, partitioning, synchronization algorithm, timing granularity

7 The LOCUS distributed operating system
Bruce Walker, Gerald Popek, Robert English, Charles Kline, Greg Thiel
October 1983 **ACM SIGOPS Operating Systems Review , Proceedings of the ninth ACM symposium on Operating systems principles**, Volume 17 Issue 5

Full text available: pdf(1.89 MB) Additional Information: full citation, abstract, references, citings, index terms

LOCUS is a distributed operating system which supports transparent access to data through a network wide filesystem, permits automatic replication of storage, supports transparent distributed process execution, supplies a number of high reliability functions such as nested transactions, and is upward compatible with Unix. Partitioned operation of subnet's and their dynamic merge is also supported. The system has been operational for about two years at UCLA a ...

8 Distributed file systems: concepts and examples
Eliezer Levy, Abraham Silberschatz
December 1990 **ACM Computing Surveys (CSUR)**, Volume 22 Issue 4

Full text available: pdf(5.33 MB) Additional Information: full citation, abstract, references, citings, index terms, review

The purpose of a distributed file system (DFS) is to allow users of physically distributed computers to share data and storage resources by using a common file system. A typical configuration for a DFS is a collection of workstations and mainframes connected by a local area network (LAN). A DFS is implemented as part of the operating system of each of the connected computers. This paper establishes a viewpoint that emphasizes the dispersed structure and decentralization of both data and con ...

9 Experience Using Multiprocessor Systems—A Status Report
Anita K. Jones, Peter Schwarz
June 1980 **ACM Computing Surveys (CSUR)**, Volume 12 Issue 2

Full text available: pdf(4.48 MB) Additional Information: full citation, references, citings, index terms

10 A flexible operation execution model for shared distributed objects
Saniya Ben Hassen, Irina Athanasiu, Henri E. Bal
October 1996 **ACM SIGPLAN Notices , Proceedings of the 11th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 31 Issue 10

Full text available: pdf(2.30 MB) Additional Information: full citation, abstract, references, citings, index terms

Many parallel and distributed programming models are based on some form of shared objects, which may be represented in various ways (e.g., single-copy, replicated, and partitioned objects). Also, many different operation *execution strategies* have been designed for each representation. In programming systems that use multiple representations integrated in a single object model, one way to provide multiple execution strategies is to implement each strategy independently from the others. How ...

11 Requirements interaction management
William N. Robinson, Suzanne D. Pawlowski, Vecheslav Volkov
June 2003 **ACM Computing Surveys (CSUR)**, Volume 35 Issue 2

Full text available: pdf(1.24 MB) Additional Information: full citation, abstract, references, index terms

Requirements interaction management (RIM) is the set of activities directed toward the discovery, management, and disposition of critical relationships among sets of requirements, which has become a critical area of requirements engineering. This survey looks at the evolution of supporting concepts and their related literature, presents an issues-based framework for reviewing processes and products, and applies the framework in a review of RIM state-of-the-art. Finally, it presents seven researc ...

Keywords: KAOS, KATE, Oz, Requirements engineering, Telos, WinWin, analysis and design, composite system, deficiency driven design, dependency analysis, distributed intentionality, interaction analysis, software cost reduction (SCR)., system architecture, system specification, viewpoints

12 Using temporal hierarchies to efficiently maintain large temporal databases

Thomas Dean

October 1989 **Journal of the ACM (JACM)**, Volume 36 Issue 4

Full text available: pdf(2.94 MB) Additional Information: full citation, abstract, references, citings, index terms

Many real-world applications involve the management of large amounts of time-dependent information. Temporal database systems maintain this information in order to support various sorts of inference (e.g., answering questions involving propositions that are true over some intervals and false over others). For any given proposition, there are typically many different occasions on which that proposition becomes true and persists for some length of time. In this paper, these occasions are re ...

13 Compile-time memory reuse in logic programming languages through update in place

Gudjón Gudjónsson, William H. Winsborough

May 1999 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 21 Issue 3

Full text available: pdf(693.38 KB) Additional Information: full citation, abstract, references, index terms

Standard implementation techniques for single-assignment languages modify a data structure without destroying the original, which may subsequently be accessed. Instead a variant structure is created by using newly allocated cells to represent the changed portion and to replace any cell that references a newly allocated cell. The rest of the original structure is shared by the variant. The effort required to leave the original uncorrupted is unnecessary when the program will never reference ...

Keywords: Prolog, compile-time garbage collection, local reuse, reuse map, update in place

14 Experiments with subdivision of search in distributed theorem proving

Maria Paola Bonacina

July 1997 **Proceedings of the second international symposium on Parallel symbolic computation**

Full text available: pdf(1.76 MB) Additional Information: full citation, references, index terms

15 On randomization in sequential and distributed algorithms

Rajiv Gupta, Scott A. Smolka, Shaji Bhaskar

March 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 1

Full text available: pdf(8.01 MB) Additional Information: full citation, abstract, references, citings, index terms

Probabilistic, or randomized, algorithms are fast becoming as commonplace as conventional deterministic algorithms. This survey presents five techniques that have been widely used in the design of randomized algorithms. These techniques are illustrated using 12 randomized algorithms—both sequential and distributed— that span a wide range of applications, including:primality testing (a classical problem in number theory), interactive probabilistic proof s ...

Keywords: Byzantine agreement, CSP, analysis of algorithms, computational complexity, dining philosophers problem, distributed algorithms, graph isomorphism, hashing, interactive probabilistic proof systems, leader election, message routing, nearest-neighbors problem, perfect hashing, primality testing, probabilistic techniques, randomized or probabilistic algorithms, randomized quicksort, sequential algorithms, transitive tournaments, universal hashing

16 Status report of the graphic standards planning committee

Computer Graphics staff

August 1979 **ACM SIGGRAPH Computer Graphics**, Volume 13 Issue 3

Full text available: pdf(15.01 MB) Additional Information: full citation, references, citings

17 A methodology and algorithms for the design of hard real-time multitasking ASICs

Miodrag Potkonjak, Wayne Wolf

October 1999 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 4 Issue 4

Full text available: pdf(198.48 KB) Additional Information: full citation, abstract, references, index terms, review

Traditional high-level synthesis concentrates on the implementation of a single task (e.g. filter, linear controller, A/D converter). However, many applications—multifunctional embedded controllers intelligent wireless end-points, and DSP and multimedia servers—are defined as sets of several computational tasks. This paper describes new techniques for the synthesis of ASIC implementations that realize multiple computational processes under hard real-time constraints. Our synthes ...

18 A declarative approach to optimize bulk loading into databases

Sihem Amer-Yahia, Sophie Cluet

June 2004 **ACM Transactions on Database Systems (TODS)**, Volume 29 Issue 2

Full text available: pdf(1.00 MB) Additional Information: full citation, abstract, references, index terms

Applications, such as warehouse maintenance, need to load large data volumes regularly. The efficiency of loading depends on the resources that are available at the source and at the target systems. Our work aims to understand the performance criteria that are involved in bulk loading data into a database and to devise tailored optimization strategies.Unlike commercial systems and previous research on the same topic, our approach follows the fundamental database principle of physical-logical ind ...

Keywords: Declarative bulk loading, algebra, recovery, side-effects

19 Concurrency and recovery in generalized search trees

Marcel Kornacker, C. Mohan, Joseph M. Hellerstein

June 1997 **ACM SIGMOD Record , Proceedings of the 1997 ACM SIGMOD international conference on Management of data**, Volume 26 Issue 2

Full text available: pdf(1.59 MB) Additional Information: full citation, abstract, references, citings, index

terms

This paper presents general algorithms for concurrency control in tree-based access methods as well as a recovery protocol and a mechanism for ensuring repeatable read. The algorithms are developed in the context of the Generalized Search Tree (GiST) data structure, an index structure supporting an extensible set of queries and data types. Although developed in a GiST context, the algorithms are generally applicable to many tree-based access methods. The concurrency control protocol is base ...

20 Concurrency and recovery for index trees

David Lomet, Betty Salzberg

August 1997 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 6 Issue 3

Full text available: pdf(168.36 KB) Additional Information: full citation, abstract, citings, index terms

Although many suggestions have been made for concurrency in B $^+$-trees, few of these have considered recovery as well. We describe an approach which provides high concurrency while preserving well-formed trees across system crashes. Our approach works for a class of index trees that is a generalization of the B $^{\rm link}$-tree. This class includes some multi-attribute indexes and temporal indexes. Structural changes in an index tree are decomposed into a sequence of atomic actions, each one ...

Keywords: Access methods, B-trees, Concurrency, Indexing, Recovery

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used **resource assignment** and **logical partition** Found **35,824** of **151,219**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Layout-driven RTL binding techniques for high-level synthesis using accurate estimators

Min Xu, Fadi J. Kurdahi

October 1997 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 2 Issue 4

Full text available: pdf(1.20 MB) Additional Information: full citation, abstract, references, citings, index terms

The importance of effective and efficient accounting of layout effects is well established in High-Level Synthesis (HLS), since it allows more realistic exploration of the design space and the generation of solutions with predictable metrics. This feature is highly desirable in order to avoid unnecessary iterations through the design process. In this article, we address the problem of layout-driven register-transfer-level (RTL) binding as this step has a direct relevance to the final perfor ...

Keywords: FPGAs, binding, floorplan, high-level synthesis

2 A multi-protocol cross-domain communication model for metacomputing systems

Bi Peng, Xie Fei, Yang Guangwen, Wang Dingxing

April 2002 **ACM SIGOPS Operating Systems Review**, Volume 36 Issue 2

Full text available: pdf(1.06 MB) Additional Information: full citation, abstract, references

Metacomputing system is a kind of heterogeneous and dynamic distributed computing system in WAN (Wide Area Network). Communication model of metasystems must tackle the heterogeneous and dynamic problems, and has the ability to pass messages in WAN. After analyzing characteristics and deficiencies in common methods, this paper proposes a new communication model for metasystems. This model is based on Java, whose platform independence and data serialization hide the heterogeneity of various nodes ...

Keywords: communication, message passing, metacomputing, parallel/distributed computing

3 An implementation of a state assignment heuristic

Alan J. Coppola

July 1986 **Proceedings of the 23rd ACM/IEEE conference on Design automation**

Additional Information:

Full text available: pdf(703.62 KB) full citation, abstract, references, citings, index terms

This paper presents the results of developing and integrating a Finite State Machine(FSM) state-assignment tool into the functional design part of the Intel PLA-based synthesis system. The tool developed is an heuristic adjacency-based state assignment program, based on the KISS program [DeMicheli 85]. Statistics are presented, relative to the tool, on 28 FSM's. The current state of the functional design tool is described, as is the interface with the new tool. A new abstra ...

4 Reconfigurable computing: a survey of systems and software

Katherine Compton, Scott Hauck

June 2002 **ACM Computing Surveys (CSUR)**, Volume 34 Issue 2

Full text available: pdf(710.56 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Due to its potential to greatly accelerate a wide variety of applications, reconfigurable computing has become a subject of a great deal of research. Its key feature is the ability to perform computations in hardware to increase performance, while retaining much of the flexibility of a software solution. In this survey, we explore the hardware aspects of reconfigurable computing machines, from single chip architectures to multi-chip systems, including internal structures and external coupling. W ...

Keywords: Automatic design, FPGA, field-programmable, manual design, reconfigurable architectures, reconfigurable computing, reconfigurable systems

5 Constraints-driven scheduling and resource assignment

Krzysztof Kuchcinski

July 2003 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 8 Issue 3

Full text available: pdf(361.41 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes a new method for modeling and solving different scheduling and resource assignment problems that are common in high-level synthesis (HLS) and system-level synthesis. It addresses assignment of resources for operations and tasks as well as their static, off-line scheduling. Different heterogeneous constraints are considered for these problems. These constraints can be grouped into two classes: problem-specific constraints and design-oriented constraints. They are uniformly mo ...

Keywords: Constraint programming, high-level synthesis, resource assignment, scheduling, system-level synthesis

6 Area and Timing Estimation for Lookup Table Based FPGAs

Min Xu, Fadi Kurdahi

March 1996 **Proceedings of the 1996 European conference on Design and Test**

Full text available: pdf(905.34 KB) Publisher Site Additional Information: full citation, abstract

The importance of efficient area and timing estimation techniques is well-established in High-Level Synthesis (HLS), since it allows more efficient exploration of the design space while providing HLS tools with the capability of predicting the effects of technology-specific tools on the design space. Much of previous work has focused on estimation techniques that use very simple cost models based solely on the gate and/or literal count. Those models are not accurate enough to allow effective des ...

7 An algorithm for improving partitions of pin-limited multi-chip systems
Mark Beardslee, Alberto Sangiovanni-Vincentelli
November 1993 **Proceedings of the 1993 IEEE/ACM international conference on Computer-aided design**
Full text available: pdf(863.47 KB) Additional Information: full citation, references

8 Effects of resource sharing on circuit delay: an assignment algorithm for clock period optimization
Subhrajit Bhattacharya, Sujit Dey, Franc Breglez
April 1998 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 3 Issue 2
Full text available: pdf(260.26 KB) Additional Information: full citation, abstract, references, index terms

This paper analyzes the effect of resource sharing and assignment on the clock period of the synthesized circuit. The assignment phase assigns or binds operations of the scheduled behavioral description to a set of allocated resources. We focus on control-flow intensive descriptions, characterized by the presence of mutually exclusive paths due to the presence of nested conditional branches and loops.We show that clustering multiple operations in the same state of the schedule, p ...

Keywords: clock period, high-level synthesis, resorce sharing

9 A configuration management approach for large workflow management systems
Hans Schuster, Jens Neeb, Ralf Schamburger
March 1999 **ACM SIGSOFT Software Engineering Notes , Proceedings of the international joint conference on Work activities coordination and collaboration**, Volume 24 Issue 2
Full text available: pdf(1.38 MB) Additional Information: full citation, abstract, references, citings, index terms

Scalability to large, heterogeneous, and distributed environments is an important requirement for workflow management systems (WfMS). As a consequence, the management of the configuration of a WfMS installation becomes a key issue. This paper proposes an approach for managing the configuration of WfMS together with an assignment strategy for workflow instances. Separating the logical issues of the workflow model from the physical configuration of a WfMS is the basis of our strategy. A formalizat ...

Keywords: configuration, scalability, workflow management system

10 Circuit partitioning with complex resource constraints in FPGAs
Huiqun Liu, Kai Zhu, D. F. Wong
March 1998 **Proceedings of the 1998 ACM/SIGDA sixth international symposium on Field programmable gate arrays**
Full text available: pdf(1.08 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper, we present an algorithm for circuit partitioning with complex resource constraints in large FPGAs. Traditional partitioning methods estimate the capacity of an FPGA device by counting the number of logic blocks, however this is not accurate with the increasing capacity and diverse resource types in the new FPGA architectures. We propose a network flow based method to optimally check whether a circuit or a sub-circuit is feasible for a set of available heterogeneous resources. ...

11 Target architecture oriented high-level synthesis for multi-FPGA based emulation
Oliver Bringmann, Carsten Menn, Wolfgang Rosenstiel
January 2000 **Proceedings of the conference on Design, automation and test in Europe**
Full text available: pdf(122.87 KB) Publisher Site Additional Information: full citation, references, index terms

12 A hybrid ASIC and FPGA architecture
Paul S. Zuchowski, Christopher B. Reynolds, Richard J. Grupp, Shelly G. Davis, Brendan Cremen, Bill Troxel
November 2002 **Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design**
Full text available: pdf(116.56 KB) Additional Information: full citation, references, citings, index terms

13 Query evaluation techniques for large databases
Goetz Graefe
June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2
Full text available: pdf(9.37 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

14 A unified approach to the decomposition and re-decomposition of sequential machines
Pranav Ashar, Srinivas Devadas, A. Richard Newton
January 1991 **Proceedings of the 27th ACM/IEEE conference on Design automation**
Full text available: pdf(854.55 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a unified framework and associated algorithms for the optimal decomposition and re-decomposition of sequential machines. This framework allows for a uniform treatment of arbitrary decomposition topologies operating at the State Transition Graph (STG) level, while targeting a cost function that is close to the eventual logic implementation. Previous work has targeted specific decomposition topologies via the formulation of decomposition as implicant covering with associated constr ...

15 Lower bounds on test resources for scheduled data flow graphs
Ishwar Parulkar, Sandeep K. Gupta, Melvin A. Breuer
June 1996 **Proceedings of the 33rd annual conference on Design automation**
Full text available: pdf(309.27 KB) Additional Information: full citation, references, citings, index terms

16 Multiple FPGA partitioning with performance optimization
Kalapi Roy-Neogi, Carl Sechen
February 1995 **Proceedings of the 1995 ACM third international symposium on Field-programmable gate arrays**
Full text available: pdf(214.18 KB) Additional Information: full citation, references, citings, index terms

17 PODEM-X: An automatic test generation system for VLSI logic structures
Prabhakar Goel, Barry C. Rosales
June 1981 **Proceedings of the 18th conference on Design automation**
Full text available: pdf(807.99 KB) Additional Information: full citation, abstract, references, citings, index terms

Multiple test generation algorithms and techniques described in this paper have been integrated into a unified system which has successfully produced tests for unpartitioned LSSD logic structures of up to 50,000 logic gates. The design concepts behind the creation of a unified system are presented, as are actual results obtained on large logic structures. System usability was significantly enhanced by the same concepts that facilitated the integration of multiple algorithms and techniques.< ...

18 Fast detection of communication patterns in distributed executions
Thomas Kunz, Michiel F. H. Seuren
November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**
Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

19 PODEM-X: An automatic test generation system for VLSI logic structures
P. Goel, B. C. Rosales
June 1988 **Papers on Twenty-five years of electronic design automation**
Full text available: pdf(1.00 MB) Additional Information: full citation, references, index terms

20 Embedded system synthesis by timing constraints solving
Krzysztof Kuchcinski
September 1997 **Proceedings of the 10th international symposium on System synthesis**
Full text available: pdf(1.14 MB) Additional Information: full citation, abstract, references, citings
Publisher Site

This paper presents an approach to embedded system synthesis which minimizes a system cost while implementing given timing requirements. The embedded system is represented by a set of finite domain constraints defining different requirements on processes timing, system resources and interprocess communication. The assignment of processes to processors and interprocess communications to buses as well as their scheduling are then defined as an optimization problem. A prototype system, based on con ...

Keywords: Embedded Systems, Synthesis, Constraint Logic Programming.

Results 1 - 20 of 200 Result page: 1 2 3 4 5 6 7 8 9 10 next

Your search matched **6** of **1140634** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

search <and>resource assignment Search

Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 **Optimal resource assignment in Internet data centers**
Xiaoyun Zhu; Singhal, S.;
Modeling, Analysis and Simulation of Computer and Telecommunication Syste 2001. Proceedings. Ninth International Symposium on , 15-18 Aug. 2001
Pages:61 - 69

[Abstract] [PDF Full-Text (680 KB)] IEEE CNF

2 **Optimizing multi-joint queries in parallel relational databases**
Srivastava, J.; Elsesser, G.;
Parallel and Distributed Information Systems, 1993., Proceedings of the Secor International Conference on , 20-22 Jan. 1993
Pages:84 - 92

[Abstract] [PDF Full-Text (792 KB)] IEEE CNF

3 **A protocol for fast resource assignment in wireless PCS**
Karol, M.J.; Chih-Lin I;
Vehicular Technology, IEEE Transactions on , Volume: 43 , Issue: 3 , Aug. 19
Pages:727 - 732

[Abstract] [PDF Full-Text (528 KB)] IEEE JNL

4 **A GA embedded dynamic search algorithm over a Petri net model fo FMS scheduling**
Yung-Feng Chiu; Li-Chen Fu;
Robotics and Automation, 1997. Proceedings., 1997 IEEE International Confer on , Volume: 1 , 20-25 April 1997
Pages:513 - 518 vol.1

[Abstract] [PDF Full-Text (604 KB)] IEEE CNF

5 **Fair deterministic packet access protocol: F-RAMA (fair resource assignment multiple access)**
Pinheiro, A.L.A.; de Marca, J.R.B.;
Electronics Letters , Volume: 32 , Issue: 25 , 5 Dec. 1996
Pages:2310 - 2311

[Abstract] [PDF Full-Text (256 KB)] IEE JNL

6 **Design Space Exploration for Dynamically Reconfigurable Architectu**
Miramond, B.; Delosme, J.-M.;
Design, Automation and Test in Europe, 2005. Proceedings , 07-11 March 200
Pages:366 - 371

[Abstract] [PDF Full-Text (168 KB)] IEEE CNF

Your search matched **0** of **1140634** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

optimize<and>logical partition<and>processor Search

Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

Results:
No documents matched your query.

Your search matched **0** of **1140634** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

resource assignment<and>persistent objects Search

Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

Results:
No documents matched your query.

Your search matched **0** of **1140634** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

persistent objects<and>logical partition Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

Results:
No documents matched your query.